Supplementary Materials for ProDepth: Boosting Self-Supervised Multi-Frame Monocular Depth with Probabilistic Fusion

Sungmin Woo*, Wonjoon Lee*, Woo Jin Kim, Dogyoon Lee, and Sangyoun Lee

Yonsei University
https://sungmin-woo.github.io/prodepth/

A Overview

This supplementary document provides additional technical details, experiments and visualization results. In Sec. B, we describe implementation details of our ProDepth including hyperparameters and training strategies. In Sec. C, we provide additional ablation study on the components of ProDepth and quantitative comparisons with related works. In Sec. D, we discuss the limitations of our work. In Sec. E, we present additional visualizations for diverse scenes.

B Implementation Details

Training. We implement our model in Pytorch [24] with two NVIDIA RTX A6000 GPUs. Following the methodology in [31], we apply color and flip augmentations to training images. Unless explicitly specified, our models take two frames $\{I_{t-1}, I_t\}$ as inputs during both training and testing, and three frames $\{I_{t-1}, I_t, I_{t+1}\}$ are used for self-supervised training. The model undergoes training for 25 epochs on Cityscapes with batch size 24 and 20 epochs on KITTI with batch size 12. We employed the Adam optimizer [18] with an initial learning rate of 10^{-4}, reduced by a factor of 10 during the final 10 epochs for Cityscapes and 5 epochs for KITTI. Pose and single-frame networks are frozen when the learning rates drop. The loss coefficients are $\lambda_1 = 1$, $\lambda_2 = 0.3$, $\lambda_3 = 0.05$, and $\lambda_s = 0.003$.

Model. The pose network uses ResNet18 [16] as an encoder, while the depth network adopts a lightweight CNN-Transformer hybrid encoder from [34]. In accordance with prior works, encoders are initialized with ImageNet [5] pretrained weights. The features employed in constructing the cost volume have a channel size of $C = 64$, with $k = 128$ hypothesized depth bins (candidates), and a binary masking threshold of $\gamma = 0.8$.

Dataset. In our study of the Cityscapes dataset, we use a set of pre-processed 58,335 training images provided by [8], along with 1,525 images for testing. For the KITTI dataset, we adhere to the Eigen split [6] following established practices [2,8,14,31]. This split encompasses 39,810 training images, 4,424 validation images, and 697 test images. For the generalization study on the Waymo Open

dataset [27], 2,216 front camera images are uniformly sampled from the validation set, which comprises 202 video sequences. In all datasets, we exclusively use unlabeled video frames, without incorporating additional segmentation masks or optical flow information. The ground-truth depth information is employed solely for evaluation, and we constrain the predicted depth values to be below 80 meters.

C Additional Experimental Results

As outlined in the main paper, our experiments primarily concentrate on the Cityscapes dataset, which features a higher number of moving objects compared to the KITTI dataset. Unless otherwise specified, all experimental results denote performance on Cityscapes.

C.1 Fusion Method for Probabilistic Cost Volume Modulation

In the proposed PCVM module, we perform an uncertainty-aware adaptive fusion of the depth probability distributions derived from single-frame and multi-frame cues in the cost volume. We explore weighted arithmetic mean (*wam*) and weighted geometric mean (*wgm*) as fusion methods. Given the probabilities $p_j \in \{p_{\text{single}}, p_{\text{cv}}\}$ and corresponding weights $w_j \in \{U, 1-U\}$, the fused probability distribution $P(d|x)$ can be obtained using *wam* (Eq. 1) or *wgm* (Eq. 2).

$$P(d|x) = \frac{\sum_j (p_j(d|x) \cdot w_j)}{\sum_j w_j} = p_{\text{single}}(d|x) \cdot U(x) + p_{\text{cv}}(d|x) \cdot (1 - U(x)). \quad (1)$$

$$P(d|x) = (\prod_j p_j(d|x)^{w_j})^{1/\sum_j w_j} = p_{\text{single}}(d|x)^{U(x)} \cdot p_{\text{cv}}(d|x)^{1-U(x)}. \quad (2)$$

As discussed in the main paper, the commonly used *wam*, with its additive nature, may not guarantee the preservation of depth candidates at the maximum due to the linear combination of distributions. It tends to alter the location of a peak (local maxima) of the distribution after fusion, where the depth candidate with the highest probability in the fused probability distribution $P(d|x)$ does not precisely represent either single-frame or multi-frame cues. However, we observe that it is more appropriate to decisively adopt one position because in most cases, the multi-frame cue is more accurate than the single-frame cue in static scenes, and vice versa in dynamic scenes. Incorporating less reliable cue with

Table 1: Fusion methods for PCVM.

Fusion Method	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Weighted Arithmetic Mean	0.098	0.945	5.715	0.152	0.898	0.974	0.992
Weighted Geometric Mean	**0.095**	**0.882**	**5.549**	**0.146**	**0.908**	**0.978**	**0.993**

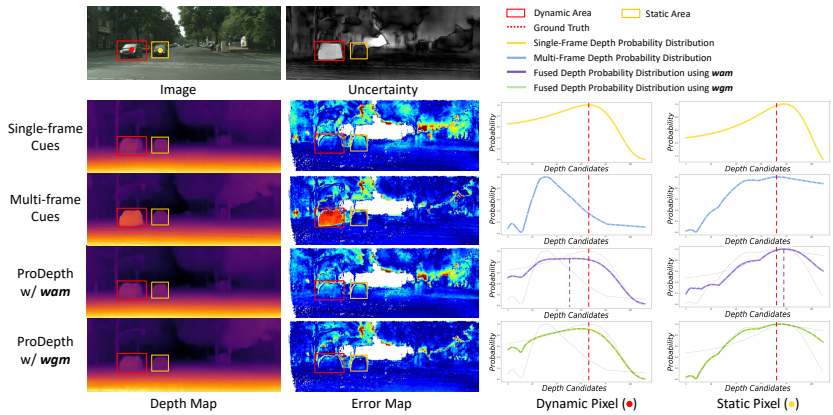

Fig. 1: Analysis on the fusion methods. The estimated depth maps, error maps, and depth probability distributions are presented. Our proposed PCVM performs uncertainty-aware adaptive fusion of probability distributions derived from single- and multi-frame cues. When the weighted arithmetic mean (*wam*) is used for fusion, the peak of the fused distribution exists between those in single- and multi-frame distributions. In contrast, when *wgm* is used for fusion, the peak of the fused distribution follows that of more reliable cues according to the inferred uncertainty.

wam may shifts the positions of peaks away from the optimal depth candidate. In contrast, *wgm* allows for the retention of depth candidates with the highest probability due to its multiplicative nature, maintaining the positions of peaks. Instead, their probabilities are adjusted with the corresponding weights. Table 1 demonstrates that *wam* degrades the performance, while *wgm* achieves superior results. Fig. 1 illustrates the analysis on the fusion methods.

C.2 Depth Evaluation on Dynamic Objects

To validate the effectiveness of our approach, we further evaluate the model's performance on dynamic objects using the Cityscapes and Waymo Open datasets.

Cityscapes Dataset. For Cityscapes dataset, we compute the depth errors within movable objects belonging to dynamic classes (*e.g.*, vehicles, pedestrians, bikes) as presented in Table 2. These objects are identified using a pretrained semantic segmentation network. While DynamicDepth [8] and InstaDM [21] utilize these segmentation masks directly in both training and inference, our ProDepth achieves the comparable performance, underscoring the effectiveness of uncertainty reasoning and probabilistic cost volume modulation. It is important to note that the evaluation involves the static objects, as segmentation does not account for their movements.

Waymo Open Dataset. As the Waymo Open dataset provides panoptic labels and 3D box positions, moving objects can be distinguished from static

Table 2: Depth errors on movable objects in dynamic classes (Cityscapes dataset).

Method	Semantics	$W \times H$	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Monodepth2 [11]		416×128	0.159	1.937	6.363	0.201	0.816	0.950	0.981
InstaDM [21]	✓	832×256	0.139	1.698	5.760	0.181	0.859	0.959	0.982
ManyDepth [31]		416×128	0.169	2.175	6.634	0.218	0.789	0.921	0.969
DynamicDepth [8]	✓	416×128	0.129	1.273	4.626	**0.168**	**0.862**	**0.965**	0.986
ProDepth w/o PCVM		416×128	0.134	1.151	4.715	0.177	0.833	0.958	0.987
ProDepth		416×128	**0.126**	**0.953**	**4.483**	0.172	0.837	0.959	**0.988**

Table 3: Generalization performance on static and dynamic areas in scenes involving moving objects (Waymo Open dataset).

Eval	Method	Semantics	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Static	ManyDepth [31]		0.259	3.770	10.018	0.320	0.590	0.849	0.932
	DynamicDepth [8]	✓	0.256	3.634	9.904	0.321	0.592	0.849	0.933
	ProDepth		**0.247**	**3.626**	**9.483**	**0.299**	**0.634**	**0.863**	**0.936**
Dynamic	ManyDepth [31]		0.376	6.661	11.559	0.381	0.498	0.757	0.879
	DynamicDepth [8]	✓	0.362	6.100	11.159	0.363	0.494	0.773	**0.900**
	ProDepth		**0.338**	**5.976**	**11.088**	**0.346**	**0.553**	**0.797**	0.898

objects by computing their motions. We derive masks for moving objects following the procedure outlined in [28], and then sample dynamic scenes containing at least one moving object. Table 3 presents the generalization performance on static and moving pixels within dynamic scenes. Our ProDepth model surpasses related approaches, benefiting significantly from PCVM, which compensates for the errors of multi-frame depth in dynamic areas. It is evident that PCVM significantly enhances performance in dynamic pixels compared to static pixels.

C.3 Additional Quantitative Results

Predictive distribution for single-frame depth estimation. The predictive distribution can be modeled as Laplace or Gaussian. As shown in Table 4, the single-frame depth represented as a Gaussian distribution slightly outperforms the Laplace distribution in conveying useful cues for probabilistic fusion in a PCVM module.

Table 4: Predictive distribution for single-frame depth estimation.

Predictive Distribution	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Laplace	0.096	0.883	5.579	**0.146**	0.907	**0.978**	**0.993**
Gaussian	**0.095**	**0.882**	**5.549**	**0.146**	**0.908**	**0.978**	**0.993**

Binary masking threshold γ. Our uncertainty-aware photometric reprojection loss $\mathcal{L}_{up}$ consists of two factors: binary masking M and loss reweighting $(1-U)$:

$$\mathcal{L}_{up} = M \odot (1-U) \odot \mathcal{L}_p, \quad M = [U < \gamma], \tag{3}$$

where $\odot$ is element-wise product and $[\cdot]$ denotes the Iverson bracket. In Table 5, we present the results obtained with various thresholds for binary masking. We adopt $\gamma = 0.8$ for the final model, which excludes dynamic areas with high uncertainty ($U > 0.8$).

Table 5: Ablation on the binary masking threshold γ.

Threshold γ	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
0.2	0.101	0.978	5.781	0.153	0.898	0.975	0.992
0.4	0.096	0.883	5.595	0.148	0.904	0.977	0.992
0.6	**0.095**	**0.869**	5.598	0.148	0.904	0.977	**0.993**
0.8	**0.095**	0.882	**5.549**	**0.146**	**0.908**	**0.978**	**0.993**

KITTI evalution on improved ground truth. In Table 6, we present the KITTI results evaluated using the improved dense ground truth [29], which is generated by accumulating 5 consecutive frames to form a denser ground truth depth map. Our approach exhibits comparable performance to the supervised method BTS [20], showcasing the effectiveness of our self-supervised multi-frame framework.

Table 6: Depth evaluation on the KITTI dataset using the improved ground truth depth maps. D indicates the depth supervision and M denotes the monocular self-supervision.

Method	Supervision	Test frames	Error metric (↓)				Accuracy metric (↑)		
			Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Kuznietsov *et al.* [19]	D	1	0.113	0.741	4.621	0.189	0.862	0.960	0.986
Gan *et al.* [10]	D	1	0.098	0.666	3.933	0.173	0.890	0.964	0.985
Guizilimi *et al.* [15]	D	1	0.072	0.340	3.265	0.116	0.934	-	-
DORN [9]	D	1	0.072	0.307	2.727	0.120	0.932	0.984	0.994
Yin *et al.* [33]	D	1	0.072	-	3.258	0.117	0.938	0.990	0.998
BTS [20]	D	1	0.059	0.245	2.756	0.096	0.956	0.993	0.998
Johnston *et al.* [17]	M	1	0.081	0.484	3.716	0.126	0.927	0.985	0.996
Packnet-SFM [13]	M	1	0.078	0.420	3.485	0.121	0.931	0.986	0.996
Monodepth2 [11]	M	1	0.090	0.545	3.942	0.137	0.914	0.983	0.995
Patil *et al.* [25]	M	N	0.087	0.495	3.775	0.133	0.917	0.983	0.995
Wang *et al.* [30]	M	2 (-1, 0)	0.082	0.462	3.739	0.127	0.923	0.984	0.996
ManyDepth [31]	M	2 (-1, 0)	0.070	0.399	3.455	0.113	0.941	0.989	0.997
DynamicDepth [8]	M	2 (-1, 0)	0.068	0.362	3.454	0.111	0.943	0.991	**0.998**
ProDepth	M	2 (-1, 0)	**0.059**	**0.308**	**3.060**	**0.097**	**0.959**	**0.992**	0.997

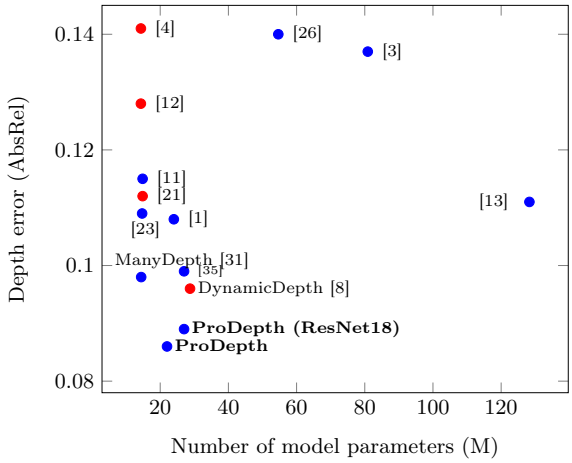

Fig. 2: Depth error on KITTI dataset against the number of model parameters. Red dots indicate models requiring semantics, and the parameters of segmentation network are not considered.

Model size and runtime. Figure 2 illustrates the depth error on the KITTI dataset plotted against the number of model parameters. Our ProDepth achieves the best performance while maintaining a comparable number of parameters. When we adopt ResNet18 [16] as the depth encoder, the performance slightly decreases while involving more parameters. ProDepth runs at 23FPS on a Titan RTX GPU.

D Limitation

Our approach is grounded in the widely accepted observation [8, 14, 22, 31, 32] that single-frame-based prediction outperforms multi-frame-based prediction in dynamic areas. However, it is important to note that single-frame estimation might struggle to achieve accurate depth for moving objects, particularly for textureless or low-light pixels, and may not offer useful cues. In addition, enabling unsupervised single-frame depth learning for dynamic regions relies on transferring knowledge from static objects, which requires a careful training strategy. The training challenges posed by datasets containing an abundance of moving objects further complicate this process.

E Additional Visualizations

We provide additional qualitative comparisons with related works [8, 31] in Figure 3 and Figure 4. Our ProDepth demonstrates accurate depth estimation, particularly in dynamic areas, highlighting the effectiveness of our probabilistic approach.

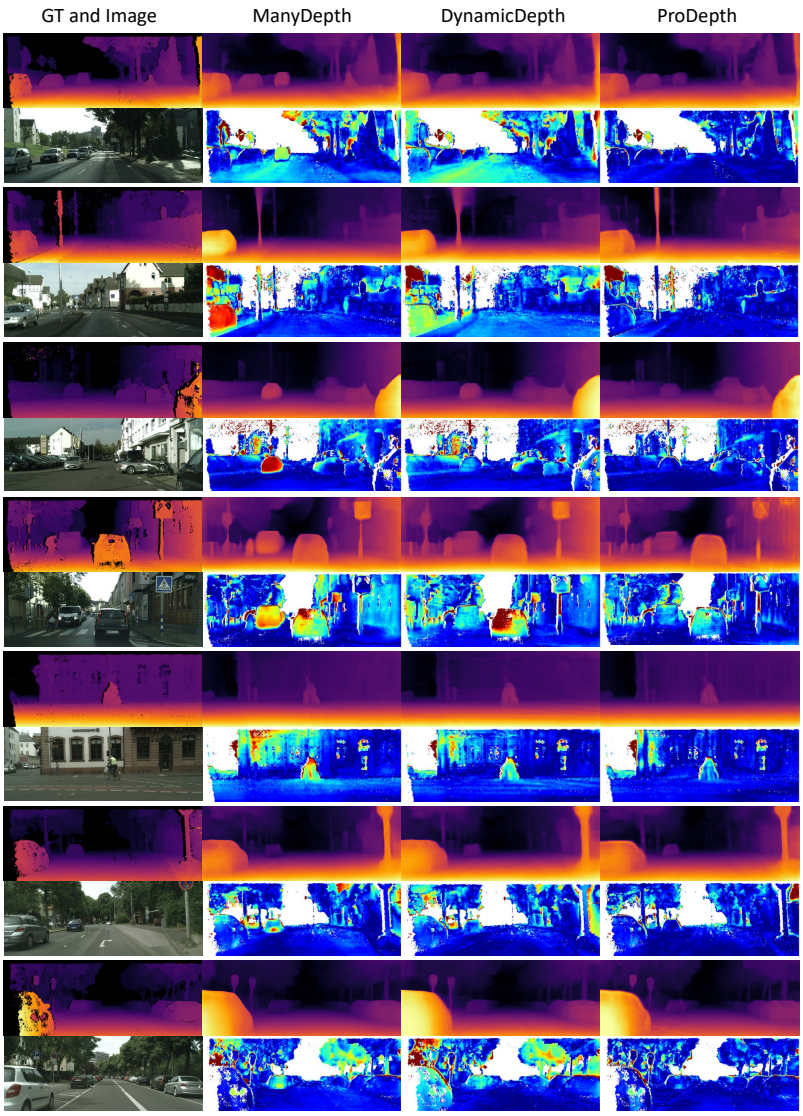

Fig. 3: Further qualitative results on the Cityscapes dataset (Part 1). Error maps in the second row for each scene measure the absolute relative error compared to the ground truth after median scaling [7], depicting large errors in red and small errors in blue.

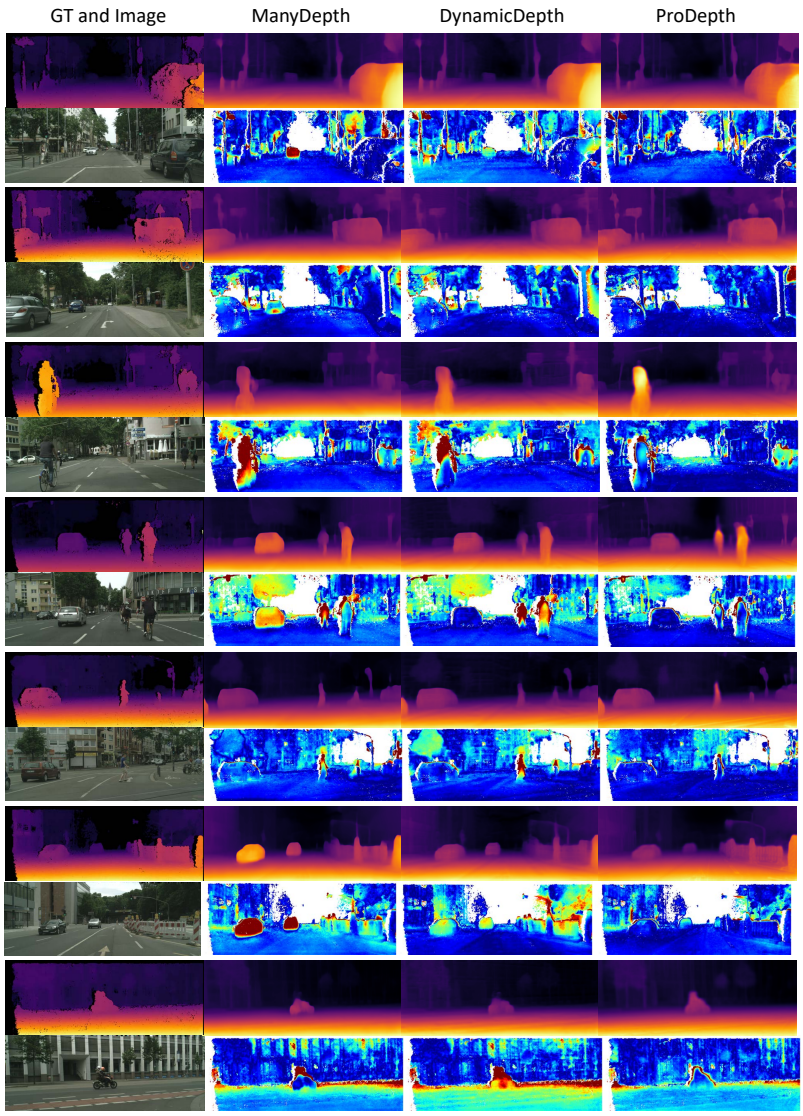

Fig. 4: Further qualitative results on the Cityscapes dataset (Part 2). Error maps in the second row for each scene measure the absolute relative error compared to the ground truth after median scaling [7], depicting large errors in red and small errors in blue.

References

1. Bae, J., Moon, S., Im, S.: Deep digging into the generalization of self-supervised monocular depth estimation. In: Proceedings of the AAAI conference on artificial intelligence. vol. 37, pp. 187–196 (2023)
2. Bangunharcana, A., Magd, A., Kim, K.S.: Dualrefine: Self-supervised depth and pose estimation through iterative epipolar sampling and refinement toward equilibrium. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 726–738 (2023)
3. Bian, J., Li, Z., Wang, N., Zhan, H., Shen, C., Cheng, M.M., Reid, I.: Unsupervised scale-consistent depth and ego-motion learning from monocular video. Advances in neural information processing systems **32** (2019)
4. Casser, V., Pirk, S., Mahjourian, R., Angelova, A.: Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 8001–8008 (2019)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
6. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE international conference on computer vision. pp. 2650–2658 (2015)
7. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE international conference on computer vision. pp. 2650–2658 (2015)
8. Feng, Z., Yang, L., Jing, L., Wang, H., Tian, Y., Li, B.: Disentangling object motion and occlusion for unsupervised multi-frame monocular depth. In: European Conference on Computer Vision. pp. 228–244. Springer (2022)
9. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2002–2011 (2018)
10. Gan, Y., Xu, X., Sun, W., Lin, L.: Monocular depth estimation with affinity, vertical pooling, and label enhancement. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 224–239 (2018)
11. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3828–3838 (2019)
12. Gordon, A., Li, H., Jonschkowski, R., Angelova, A.: Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8977–8986 (2019)
13. Guizilini, V., Ambrus, R., Pillai, S., Raventos, A., Gaidon, A.: 3D packing for self-supervised monocular depth estimation. In: CVPR (2020)
14. Guizilini, V., Ambruș, R., Chen, D., Zakharov, S., Gaidon, A.: Multi-frame self-supervised depth with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 160–170 (2022)
15. Guizilini, V., Li, J., Ambrus, R., Pillai, S., Gaidon, A.: Robust semi-supervised monocular depth estimation with reprojected distances. In: Conference on robot learning. pp. 503–512. PMLR (2020)

16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
17. Johnston, A., Carneiro, G.: Self-supervised monocular trained depth estimation using self-attention and discrete disparity volume. In: CVPR (2020)
18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Kuznietsov, Y., Stuckler, J., Leibe, B.: Semi-supervised deep learning for monocular depth map prediction. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6647–6655 (2017)
20. Lee, J.H., Han, M.K., Ko, D.W., Suh, I.H.: From big to small: Multi-scale local planar guidance for monocular depth estimation. arXiv preprint arXiv:1907.10326 (2019)
21. Lee, S., Im, S., Lin, S., Kweon, I.S.: Learning monocular depth in dynamic scenes via instance-aware projection consistency. In: 35th AAAI Conference on Artificial Intelligence/33rd Conference on Innovative Applications of Artificial Intelligence/11th Symposium on Educational Advances in Artificial Intelligence. pp. 1863–1872. ASSOC ADVANCEMENT ARTIFICIAL INTELLIGENCE (2021)
22. Li, R., Gong, D., Yin, W., Chen, H., Zhu, Y., Wang, K., Chen, X., Sun, J., Zhang, Y.: Learning to fuse monocular and multi-view cues for multi-frame depth estimation in dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21539–21548 (2023)
23. Lyu, X., Liu, L., Wang, M., Kong, X., Liu, L., Liu, Y., Chen, X., Yuan, Y.: Hr-depth: High resolution self-supervised monocular depth estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 2294–2301 (2021)
24. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32** (2019)
25. Patil, V., Van Gansbeke, W., Dai, D., Van Gool, L.: Don't forget the past: Recurrent depth estimation from monocular video. IEEE Robotics and Automation Letters **5**(4), 6813–6820 (2020)
26. Ranjan, A., Jampani, V., Kim, K., Sun, D., Wulff, J., Black, M.J.: Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In: CVPR (2019)
27. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)
28. Sun, Y., Hariharan, B.: Dynamo-depth: Fixing unsupervised depth estimation for dynamical scenes. Advances in Neural Information Processing Systems **36** (2024)
29. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity invariant cnns. In: 2017 international conference on 3D Vision (3DV). pp. 11–20. IEEE (2017)
30. Wang, J., Zhang, G., Wu, Z., Li, X., Liu, L.: Self-supervised joint learning framework of depth estimation via implicit cues. arXiv:2006.09876 (2020)
31. Watson, J., Mac Aodha, O., Prisacariu, V., Brostow, G., Firman, M.: The temporal opportunist: Self-supervised multi-frame monocular depth. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1164–1174 (2021)

32. Wimbauer, F., Yang, N., Von Stumberg, L., Zeller, N., Cremers, D.: Monorec: Semi-supervised dense reconstruction in dynamic environments from a single moving camera. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6112–6122 (2021)
33. Yin, W., Liu, Y., Shen, C., Yan, Y.: Enforcing geometric constraints of virtual normal for depth prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5684–5693 (2019)
34. Zhang, N., Nex, F., Vosselman, G., Kerle, N.: Lite-mono: A lightweight cnn and transformer architecture for self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18537–18546 (2023)
35. Zhao, C., Zhang, Y., Poggi, M., Tosi, F., Guo, X., Zhu, Z., Huang, G., Tang, Y., Mattoccia, S.: Monovit: Self-supervised monocular depth estimation with a vision transformer. In: 2022 international conference on 3D vision (3DV). pp. 668–678. IEEE (2022)